

高等职业教育

计算机类专业 规划教材

INFORMATION TECHNOLOGY

Java程序设计 案例教程

主 编 印 梅
副主编 张金学 王 莹 邢志斌
主 审 苏宝莉



中国电力出版社

<http://jc.cepp.com.cn>

计算机组成原理（第3版）

ASP.NET动态网站开发项目教程

ASP.NET动态网站开发案例教程

Visual Basic.NET程序设计项目教程

Java程序设计项目教程

SQL Server 2005数据库技术项目教程

SQL Server 2005数据库技术案例教程

软件开发实训

管理信息系统设计与开发

管理信息系统（ERP）实施

Photoshop图形图像处理项目教程

网络动画设计与制作项目教程

Linux应用服务器构架案例教程

计算机网络系统集成项目教程

Dreamweaver+ASP动态网站设计与开发项目教程

局域网组建与维护项目教程

Windows 2003 Server管理与应用项目教程

Java程序设计案例教程

计算机网络技术项目化教程

网页设计与制作项目教程

ISBN 978-7-5083-8267-8



9 787508 382678 >

定价：20.00元

高等职业教育

计算机类专业 规划教材

INFORMATION TECHNOLOGY

Java程序设计 案例教程

主 编 印 梅

副主编 张金学 王 莹 邢志斌

主 审 苏宝莉



中国电力出版社

<http://jc.cepp.com.cn>

内容提要

本书为高等职业教育计算机类专业规划教材。

本书以案例的形式,全面讲述了Java面向对象的编程。从JDK的安装和配置、Myeclipse集成开发环境的使用开始,循序渐进、由浅入深,用生动的案例介绍了Java的基本数据类型、数组、字符串、变量、方法、控制结构、类、对象、构造方法、方法重载、方法覆盖,同时以面向对象的三大原则——封装、继承、多态为主线,讲述了面向对象的高级编程;另外还讲述了Java异常处理机制、图形用户界面编程、文件和流、数据库编程和多线程。读者可以通过学习,逐步掌握Java的编程技术。

本书可作为高等职业技术学院、高等专科学校,成人高校,本科院校举办的二级职业技术学院计算机相关专业的教材,也可作为欲学习Java编程人员的参考用书。

图书在版编目(CIP)数据

Java 程序设计案例教程 / 印梅主编. —北京:中国电力出版社, 2008

高等职业教育计算机类专业规划教材

ISBN 978-7-5083-8267-8

I. J… II. 印… III. JAVA 语言—程序设计—高等学校:技术学校—教材 IV. TP312

中国版本图书馆CIP数据核字(2008)第213954号

丛 书 名: 高等职业教育计算机类专业规划教材

书 名: Java 程序设计案例教程

出版发行: 中国电力出版社

地 址: 北京市三里河路6号

邮政编码: 100044

电 话: (010) 68362602

传 真: (010) 68316497, 88383619

服务电话: (010) 58383411

传 真: (010) 58383267

E-mail: infopower@cepp.com.cn

印 刷: 北京丰源印刷厂

开本尺寸: 184mm×260mm 印 张: 12.75 字 数: 282千字

书 号: ISBN 978-7-5083-8267-8

版 次: 2008年12月北京第1版

印 次: 2008年12月第1次印刷

印 数: 0001—3000册

定 价: 20.00元

敬告读者

本书封面贴有防伪标签,加热后中心图案消失

本书如有印装质量问题,我社发行部负责退换

版权专有 翻印必究

前言

随着计算机技术在各个领域的广泛应用以及计算机系统平台和软件开发平台的不断更新,人们对程序设计语言的要求越来越高,希望程序设计语言能够方便软件开发,所开发的软件能在各种系统平台上正常运行并能很好地应用于互联网。作为一种程序设计语言,Java 具备了一些独特的优点。例如,Java 具有分布式的、面向对象的、不依赖于计算机系统的结构,这样所开发的软件就具备了跨平台性和可移植的功能。随着互联网的快速发展及网络时代的到来,各种平台的系统相互连接,Java 与平台无关的优势得到了充分的体现,由 Java 编写的程序可在各种平台上使用,避免了软件重复开发。

本书改变了以往编程语言书籍以理论为导向的格局,通过案例为先导,使得学生在直观感受的前提下,对照给出的案例,学习相应的理论知识。同时,在讲解知识点的过程中,穿插了对案例代码的解释,以及突出解释了新知识点在案例中的实现来组织内容。

第 1 章简单介绍了 Java,包括 Java 的产生和发展、Java 的特点、Java 开发环境的安装与配置,同时给出了 Java 应用程序和 Java Applet 的最简单案例,给读者一个最直观的感受,并解释了 Java 程序的组成元素;最后介绍了现今较为流行的集成开发环境 Eclipse 的使用,本章所有的程序都是在该环境下编写的。

第 2 章介绍了 Java 的语言基础,包括标识符和关键字、基本数据类型、变量和常量、算术运算符和表达式、数据类型之间的转换和控制结构等。

第 3 章重点介绍了数组的应用,包括数组的声明、创建和初始化等。

第 4 章介绍了字符串的简单应用,包括字符串的创建、操作、连接和转换等,着重强调了与 C 或 C++语言中字符串的区别。

第 5 章介绍了面向对象编程的一些关键概念,包括类和对象的创建、对象变量和方法的引用、方法的重载、访问控制符以及包的定义和使用。

第 6 章在第 5 章的基础上,介绍了面向对象编程的一些高级编程的概念,包括封装、继承、多态以及抽象类和接口。

第 7 章介绍了 Java 的异常处理,重点介绍了异常处理过程中用到的五个关键字,try、catch、finally、throw、throws,另外还介绍了如何创建自定义异常,以及异常的分类。

第 8 章介绍了如何创建图形用户界面,包括容器、布局管理、常用的基本组件。

第 9 章介绍了 GUI 事件处理机制,包括事件、事件处理机制的简介、事件适配器以及 Java 内部类在事件处理代码中的应用。

第 10 章介绍了 Java 的输入输出处理,包括字符流、字节流的输入输出,另外还讲述了文件 File 类。

第 11 章介绍了数据库的编程技术,包括 JDBC 的概述、常见的数据库驱动、MySQL

数据库的简单介绍以及如何用代码实现与数据库的连接,并实现增、删、改、查等操作。

第12章介绍了多线程的相关知识以及代码实现,包括线程的同步与线程间的通信。

本书配有与教材配套的授课课件,所有案例的源代码,以及相关资源,如有需要,可发邮件至 yinmei1983@yahoo.com.cn。

本书由印梅主编,张金学、王莹、邢志斌副主编。其中第1~3章由王莹编写,第4~7章由印梅编写,第8~10章由邢志斌编写,第11~12章由张金学编写。全书由张金学统稿,苏宝莉主审。

本书可作为高等职业技术学院、高等专科学校,成人高校,本科院校举办的二级职业技术学院计算机相关专业的教材,也可作为欲学习Java编程的用户的参考用书。

由于时间仓促、作者水平有限,在书的结构和内容上难免有不妥之处,恳请各位同行和广大读者给予批评指正。

作者

2008年11月

目 录

前 言	
第 1 章 Java 简介	1
1.1 Java 的产生与发展	1
1.2 Java 的特点	1
1.3 Java 开发环境的安装与配置	3
1.4 简单程序的编写 (application 与 applet)	4
1.5 常见集成开发环境简介	7
小结	10
习题	10
第 2 章 Java 语言基础	11
2.1 标识符与关键字	11
2.2 基本数据类型	12
2.3 变量与常量	13
2.4 算术运算符与表达式	14
2.5 布尔数据类型及其运算	15
2.6 数据类型转换	16
2.7 选择结构语句	17
2.8 循环结构语句	18
2.9 break 与 continue 语句	20
2.10 return 语句	22
2.11 递归调用方法	23
小结	24
习题	25
第 3 章 数组	26
3.1 数组声明、创建和初始化	26
3.2 数组数据类型与普通数据类型的区别	27
3.3 数组与方法调用	29
3.4 二维数组	32
小结	34
习题	34

第 4 章 简单字符串	36
4.1 字符串的创建	36
4.2 对字符串的操作	37
4.3 字符串的连接	39
4.4 字符串转换	40
小结	41
习题	41
第 5 章 面向对象的编程	43
5.1 面向对象编程	43
5.2 类的创建	44
5.3 对象的创建以及使用	47
5.4 构造方法	49
5.5 方法重载	52
5.6 static 关键字	53
5.7 包	54
5.8 访问控制权限	56
5.9 StringBuffer 类	60
5.10 内部类	61
小结	63
习题	63
第 6 章 面向对象的高级编程	64
6.1 封装	64
6.2 继承以及方法的覆盖	66
6.3 多态	69
6.4 抽象类	71
6.5 接口	73
小结	76
习题	76
第 7 章 异常处理	78
7.1 异常的介绍和分类	78
7.2 try-catch-finally	79
7.3 throw 关键字	83
7.4 throws 关键字	84
7.5 自定义异常	85
7.6 Java 的内置异常的分类	87
小结	89
习题	89

第 8 章 创建用户界面	91
8.1 容器	91
8.2 布局管理器	98
8.3 常用基本组件	101
小结	108
习题	109
第 9 章 GUI 事件处理机制	110
9.1 事件与事件处理机制	110
9.2 事件适配器	114
9.3 内部类	117
9.4 事件的取舍	120
小结	126
习题	126
第 10 章 输入和输出	128
10.1 Reader 与 Writer 字符流	128
10.2 InputStream 和 OutputStream 字节流	131
10.3 DataInputStream 和 DataOutputStream 字节流	135
10.4 随机存取文件	137
10.5 键盘输入	140
10.6 File 类以及文件读写	142
小结	144
习题	144
第 11 章 数据库编程技术	146
11.1 JDBC 概述	146
11.2 通过 JDBC 访问各种数据库的方法	147
11.3 MySQL 数据库简介	148
11.4 JDBC driver for MySQL	150
11.5 SQL	151
11.6 综合实例	154
小结	160
习题	160
第 12 章 多线程	162
12.1 线程和多线程	162
12.2 线程的实现	163
12.3 扩展 Thread 类创建线程	163
12.4 实现 Runnable 接口创建线程	165
12.5 多线程的实现	166

12.6	线程状态与生命周期.....	168
12.7	线程的同步.....	169
12.8	线程间的通信.....	171
	小结.....	174
	习题.....	174
	附录 A.....	175
	参考文献.....	194
10.1	输入和输出.....	10.1
10.2	Reader 与 Writer 接口.....	10.2
10.3	InputStream 和 OutputStream 接口.....	10.3
10.4	DataInputStream 和 DataOutputStream 接口.....	10.4
10.5	随机存取文件.....	10.5
10.6	缓冲流.....	10.6
10.7	File 类以及文件流.....	10.7
10.8	小结.....	10.8
10.9	习题.....	10.9
11.1	数据库编程技术.....	11.1
11.2	JDBC 概述.....	11.2
11.3	通过 JDBC 访问数据库的基本步骤.....	11.3
11.4	MySQL 数据库简介.....	11.4
11.5	JDBC driver for MySQL.....	11.5
11.6	SQL.....	11.6
11.7	综合实例.....	11.7
11.8	小结.....	11.8
11.9	习题.....	11.9
12.1	多线程.....	12.1
12.2	线程的创建.....	12.2
12.3	线程的调度.....	12.3
12.4	线程的同步.....	12.4
12.5	线程的通信.....	12.5

第 1 章 Java 简介

【本章介绍】

你可能听说过 Java 并急于学习 Java 语言。本章就是学习 Java 的入门阶梯，对 Java 的主要特征进行了概括，例如：面向对象编程、独立于平台、Java 字节代码、安全、性能、多线程和网络。本章也介绍了如何创建、编译并运行 Java 应用程序和 applet，提供了编写应用程序的简单例子，以及对编程结构的简短剖析。

【本章单词】

.Java file

.class file

Comment

Compile

Java command

Javac command

Java Development Toolkit (JDK)

Main method

Source code

Source file

1.1 Java 的产生与发展

Java 是 James Gosling 领导的小组在 Sun Microsystems 公司开发的，原来叫 Oak，是 1991 年为了移植入消费类电子产品而设计的。1995 年改名为 Java，用于开发 Internet 应用程序。Java 是功能齐全的程序设计语言，可以开发可靠的应用程序。现在，Java 不仅仅用于 Web 程序设计，还应用于服务器、台式机、移动设备上开发跨平台的独立应用程序。

1.2 Java 的特点

Java 语言应用非常广泛，这要归功于它的设计和程序特征。Sun 公司在 Java 语言白皮书上说：Java 是简单的 (simple)、面向对象的 (object-oriented)、分布式的 (distributed)、解释型的 (interpreted)、健壮的 (robust)、安全的 (secure)、结构中立的 (architecture-neutral)、可移植的 (portable)、高效的 (high-performance)、多线程的 (multithreaded)、动态的

(dynamic)。下面简单了解一下这些词汇的含义。

1. 简单性

Java 语言是一种面向对象的语言，它通过提供最基本的方法来完成指定的任务，只需理解一些基本的概念，就可以用它编写出适合于各种情况的应用程序。Java 略去了运算符重载、多重继承等模糊的概念，并且通过实现自动垃圾收集大大简化了程序设计者的内存清理工作。另外，Java 也适合于在小型机上运行。

2. 面向对象

Java 语言的设计集中于对象及其接口，它提供了简单的类机制以及动态的接口模型。对象中封装了它的状态变量以及相应的方法，实现了模块化和信息隐藏；而类则提供了一类对象的原型，并且通过继承机制，子类可以使用父类所提供的方法，实现了代码的复用。

3. 分布式

Java 是面向网络的语言。通过它提供的类库可以处理 TCP/IP 协议，用户可以通过 URL 地址在网络上很方便地访问其他对象。

4. 解释执行

Java 解释器直接对 Java 字节码进行解释执行。字节码本身携带了许多编译信息，使得连接过程更加简单。

5. 健壮性

Java 在编译和运行程序时，都要对可能出现的问题进行检查，以消除错误的产生。它提供自动垃圾收集来进行内存清理，防止程序员在治理内存时轻易产生错误。通过集成的面向对象的异常处理机制，在编译时，Java 提示可能出现但未被处理的异常，帮助程序员正确地进行选择以防止系统的崩溃。另外，Java 在编译时还可捕捉类型声明中的许多常见错误，防止动态运行时出现不匹配问题。

6. 安全性

用于网络、分布环境下的 Java 必须要防止病毒的入侵。Java 不支持指针，一切对内存的访问都必须通过对象的实例变量来实现，这样就防止程序员使用“特洛伊”木马等欺骗手段访问对象的私有成员，同时也避免了指针操作中容易产生的错误。

7. 体系结构中立

Java 解释器生成与体系结构无关的字节码指令，只要安装了 Java 运行系统，Java 程序就可在任意的处理器上运行。这些字节码指令对应于 Java 虚拟机中的表示，Java 解释器得到字节码后，对它进行转换，使之能够在不同的平台运行。

8. 可移植性

也称平台无关性，它使 Java 程序可以方便地被移植到网络上的不同机器。同时，Java 的类库中也实现了与不同平台的接口，使这些类库可以移植。另外，Java 编译器是由 Java 语言实现的，这使得 Java 系统本身也具有可移植性。

9. 高效性

和其他解释执行的语言如 BASIC 不同，Java 字节码的设计使之能很轻易地直接转换

成对应于特定 CPU 的机器码，从而得到较高的性能。

10. 多线程 (application) 多线程的简单应用

多线程机制使应用程序能够并行执行，而且同步机制保证了对共享数据的正确操作。通过使用多线程，程序设计者可以分别用不同的线程完成特定的行为，而不需要采用全局的事件循环机制，这样就很容易地实现网络上的实时交互行为。

11. 动态性

Java 的设计使它适合于一个不断发展的环境。在类库中可以自由地加入新的方法和实例变量而不会影响用户程序的执行。并且 Java 通过接口来支持多重继承，使之比严格的类继承具有更灵活的方式和扩展性。

1.3 Java 开发环境的安装与配置

JDK (Java Development Kit, Java 开发包, Java 开发工具) 是 Java 的 applet 和应用程序的程序开发环境。它由一个处于操作系统层之上的运行环境和开发环境，调试和运行 applet 和应用程序所需的工具组成。下面介绍 JDK 的安装和配置。

首先运行 JDK，我们以 J2SDK1.5.0 (可以在 Java 官方网站上免费下载) 为例。下载安装文件后完成安装。下面是配置过程：

- (1) 找到桌面上的“我的电脑”图标，单击右键，找到属性栏，选中。
- (2) 选择“高级”选项卡里的“环境变量”。
- (3) 在“用户变量配置”中找出“path”，如果没有，则新建一个。
- (4) 单击“path”，选择“编辑”，加入 bin 文件夹的路径。
- (5) 单击“确定”按钮，完成配置。
- (6) 单击“开始”→“运行”命令，键入“cmd”，按回车键；再键入“javac”，看出现的提示，如果提示“无效命令”，则没有配置成功，否则已经成功。

环境配置如图 1-1 所示。

寻找 Java 可执行文件目录如图 1-2 所示。

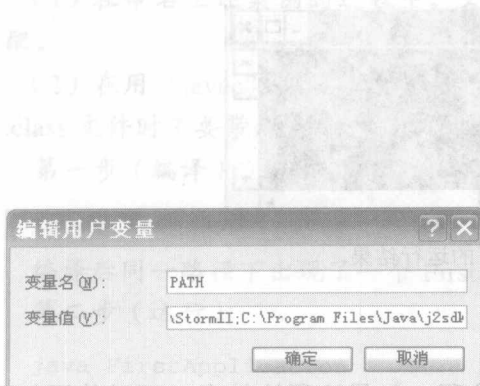


图 1-1 环境配置

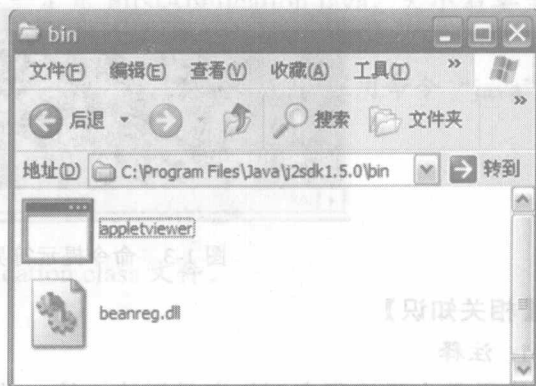


图 1-2 寻找 Java 可执行文件目录

1.4 简单程序的编写（application 与 applet）

安装完 JDK，并且配置了正确的路径之后就可以编写 Java 程序了。Java 程序分为 Java 应用程序（Java application）和 Java 小程序（Java applet）。下面学习两种程序，观察区别。

【案例 1-1】 第一个 Java application。

【学习目标】

编写 Java application 程序，学习如何用 JDK 进行编译和运行，了解 application 的特点。

【案例描述】

能够向屏幕输出“Hello World!”的 Java 应用程序。

【操作步骤】

1. 建立 java 文件

在 D 盘中建立文件夹 MyJava，在里面用编辑器建立一个文件，命名为 FirstApplication.java。

```
1. //我的第一个 java 应用程序
2. public class FirstApplication
3. {
4.     public static void main(String[] args)
5.     {
6.         System.out.println("Hello World!");
7.     }
8. }
```

2. 编译和运行

在“开始”菜单中单击“运行”命令，输入 cmd，按回车键，进入到命令提示符界面。

用“javac”命令将 FirstApplication.java 文件编译成 FirstApplication.class 文件；再用“java”命令运行 FirstApplication.class。

命令提示符界面下的运行结果如图 1-3 所示。

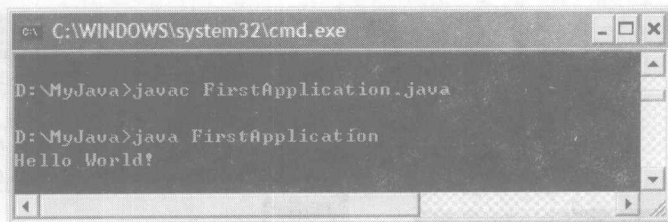


图 1-3 命令提示符界面下的运行结果

【相关知识】

1. 注释

注释是为了让程序更加容易阅读，第 1 句是注释，如果注释比较多，可以使用如下方法：

```
/*
...多行注释...
*/
```

2. 关键字

关键字是指对编译器有特殊意义，在程序中不能用作其他目的的字。比如 `class` 告诉编译器将会出现一个类。`public`、`static`、`void` 也是保留字，它们的用途以后介绍。

3. 修饰符

修饰符是特定的保留字，用来指定数据、方法、类的属性和使用方式。这里的 `public`、`static` 就是修饰符。

4. 语句

语句表示一个操作或一系列操作。比如第 6 句，表示打印“Hello World!”，语句用“;”结束。

5. 类

类（`class`）是 Java 最基本的概念，类是 Java 程序的基础。第 2～第 8 句就是一个类，名字叫 `FirstApplication`。另外要注意它是 `public` 类，所以本文件的名字一定也是 `FirstApplication.java`，否则无法编译。

6. 方法

Java 中的方法类似于 C 中的“函数”。Java 中称为方法（`method`），方法只能存在于类（`class`）中。第 4～第 7 句是一个方法。

7. 块

程序中某些成分组合构成一个块。Java 中以 `{.....}` 划分一个块，用来将一个方法、一个类划分出来。

8. main 方法

每个 Java application 必须有一个 `main` 方法，程序从 `main` 开始运行。第 4～第 7 句就是一个 `main` 方法。

【提示】

（1）在命名上述案例的过程中，文件名一定是 `FirstApplication.java`，大小写要完全匹配。

（2）在用“`javac`”命令编译原文件时，记得文件要带后缀，可是在用命令“`java`”运行 `.class` 文件时不要带后缀。例如：

第一步（编译）

```
javac FirstApplication.java
```

编译后同一路径下出现了一个 `FirstApplication.class` 文件。

第二步（运行）

```
java FirstApplication
```

运行的是 `FirstApplication.class` 文件，运行后出现结果。

【案例 1-2】 第一个 Java applet。**【学习目标】**

编写 Java applet 程序，学习如何用 JDK 进行编译和运行；了解 applet 的特点。

【案例描述】

写一个能输出 “the first java little program” 的 Java 小应用程序 (Applet)。

【操作步骤】

1. 建立 java 文件

在 D 盘中建立文件夹 MyJava，在里面用写字本建立一个文件，命名为 FirstApplet.java。

```
1. import java.applet.*;
2. import java.awt.*;
3. public class FirstApplet extends Applet {
4.     public void paint(Graphics g){
5.         g.drawString("the first java little program",10,10);
6.     }
7. }
```

2. 在同一文件夹里建立 applet.html

```
1. <html>
2. <head>
3. <title>网页</title>
4. </head>
5. <body>
6. <applet
7. code="FirstApplet.class"
8. width=200
9. height=200>
10. </applet>
11. </body>
12. </html>
```

3. 编译和运行

从开始菜单中找到 “运行” 命令，单击，输入 “cmd”，按回车键，进入到命令提示符界面，如图 1-4 所示。

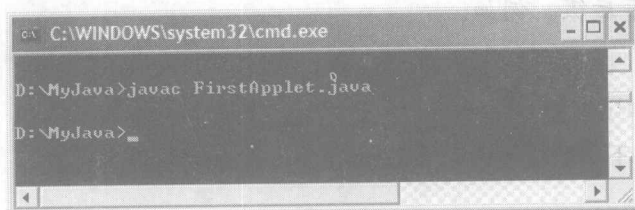


图 1-4 编译 applet

打开 applet.html，applet 运行效果如图 1-5 所示。